Zero Retries

Newsletter of NAPRA

Northwest Amateur Packet Radio Association

December 1987

Meetings

NAPRA general membership meetings are held on the second Saturday of each month at 1:00 pm. The location of the meeting alternates between the Radio Club of Tacoma's clubhouse and the clubhouse of the BEARS in Renton. Odd numbered month's meetings are held in Tacoma and even ones in Renton. The next meeting will be on December 12 at 1:00 pm in Renton and the first meeting of 1988 will be on January 9 in Tacoma. Meetings of NAPRA's board of directors are held at the Omni Restaurant near Sea-Tac Airport on the third Wednesday of each month. Both the general membership meetings and the board meetings are open to all.

Dues

Please look at your address labels to see when your NAPRA membership dues are due to be paid. Many members' dues are due in December. Please pay your dues promptly and encourage others to join NAPRA.

President's Comments

Today packets move quickly along a 220 MHz backbone linking nearly everycorner of Washington State and parts of Canada and Oregon. This backbone has been a long time goal of many people. Building it has taken a couple of years. The introduction of NET/ROM by software 2000 Inc. knocked out the last big obstacle that had challenged our best technicians. Its release last spring was timely, and NAPRA was ready. We already had the equipment and many of the mountain sites. The plan had been in existence for a long time, NET/ROM made it possible. Now we look to the future and ponder things to come. The job isn't done, the reality of routing problems and linking congestion are still here. Those of you who experienced packet before NET/ROM, can appreciate the advances that have been made, and probably don't miss the all too frequent ***RETRY COUNT EXCEEDED.

The fact is, the congestion that burdened extensive 5.01 networks still exists. It has not yet regained the magnitude that we experience so often, but nonetheless, it is growing and will effect all of us again. This congestion has been moved away from the screens and receivers of we packet users, and onto a backbone frequency that is monitored by very few.

Many parts of the country are currently trying to model packet networks in the style of ours. Here each Local Area Network enjoys RF isolation from its neighbors. Through coordination and cooperation we have been able to construct an array of LANs capable of serving nearly 200 hams across the state who can connect with each other even though the packeteers are spread across nearly half a dozen two meter frequencies. This works because each LAN has a linking node controller operating on a common 220 frequency.

Just as early packeteers witnessed on 5.01, more and more stations are popping up on the packet backbone. These are all linking nodes funneling QSOs in and out of Washington, across the Cascades, and into every neighboring state and province. We are moving the mess that was once on two meters to another frequency where it could go unnoticed.

In anticipation of this problem, it is our goal to see to it that the linking backbone does not degenerate to a useless channel filled with QRM and clutter. To do so effectively we are actively pursuing both technical and political solutions.

Experiments that were conducted a few years ago will be recreated on a wide scale on the backbone. By installing central 440 MHz transmitters at key locations along the backbone, we hope to "mark the channel busy" whenever a linking node within range of the backbone hub transmits. With this technique we can cut down on collisions that occur when two nodes separated by distance and mountains key up at the same time leaving the node in the middle unable to decode either of them.

To be sure that the backbone does not simply collapse under the weight of too many linking nodes, we are approaching the WWARA, for assistance on coordinating transmitters that operate in this frequency. Although the practice has already drawn some criticism, we must limit the number of nodes on any given backbone. It is much like the occupancy limit on an elevator. It seems unfair to tell someone that there isn't room for even one more, but the safe and efficient operation of the equipment requires us to ask some folks to take another lift.

We are seriously looking into providing the state with more than one linking frequency so that LAN to LAN traffic can move in two different directions at the same time.

It is in these directions we begin the fourth year

of NAPRA. In the past, your support and enthusiasm has made it possible to provide excellent packet service to hams in the Pacific Northwest. We hope that that support will continue as we undertake these new challenges.

Budd Churchward, WB7FHC
President NAPRA

Tomorrow...

This is an exciting period of Packet radio growth in the Pacific Northwest! The hilltop Digis are being replaced with Net/Rom based TNC's, and there is talk of high speed packet being just around the corner (see "GROWTH"). Also via the Amateur Radio Packet medium is the development of LAN's and access to new areas such as Eastern Wa., Idaho, Oregon and Vancouver, B.C. Many changes have taken place in packet radio in the past six months! In an ongoing attempt to keep you all up to date on packet happenings, future editions of "Zero Retries" will address such topics as getting started in Packet Radio, TCP/IP protocol, high speed packet technology, LAN development, special interest LANs (such as the Youth Net), packet network maps, BBS operation and traffic forwarding, packet radio and the Nat'l. Traffic System (NTS) operation, packet radio and Emergency services and others. If you have an idea for an article of interest or feel an article covering some other area of packet radio would be helpful to the packet community, please let me know. Along those same lines, there are a few packeteers in our midst who have purchased some of the newer high speed modems to experiment with, so, you can expect to be hearing from them in the near future. (News note: Hardware was received in early Nov.)

Our Technical VP, Bob Donnell , KD7NM, has written an excellent article entitled "Growth". It is a must-read for all of us. Bob's well thought out view of the future of packet radio for the Northwest has been long in coming and is welcomed . Thanks Bob, for your "looking down the lines"

We're looking forward to receiving items of news from our Eastern Wa., Vancouver B.C. and Oregon packet groups that can be included in future issues.

The NAPRA board has made a commitment to keep you informed of the progress and happenings in packet radio via Zero Retries and the Thursday night Packet Radio Ragchew. Currently Zero Retries is scheduled for publication every other month and the Packet Radio Ragchew will continue on Thursday evening at 2100 hrs. on the BEARS Repeater, 145.33 Mhz.

We look forward to seeing you at the meetings and hearing you on the Ragchew.

Our special thanks to Peter Dahl, WA7FUS, for being our interim editor for Zero Retries. Which leads me to ask if there is anyone out there that is interested in overseeing the publishing of Zero Retries every other month. Please let Peter know.

Along those same lines, the Thursday night Packet Radio Ragchew would like a few volunteers to be net-control operators. This opportunity is for one Thursday evening a month. Any volunteers for either task can expect the full cooperation and support from NAPRA. You are not expected to do it all by yourself. (I licked the stamps on the last issue of Zero Retries!)
de Don, KA6QMT

What's TCP/IP, and why should I care?
by Clifford Neuman (N1DMM)

In the past year there has been lots of talk of various "layer 3" protocols for packet radio. The first question that probably came to mind is "What's layer 3, and how is it different than what I'm using now?" A second question that you should be asking is "What is TCP/IP, and how does it relate to layer 3?" In this article, I will try to answer both these questions.

Layer 3 is referred to as such because it is the third layer in the International Standards Organization model for Open Systems Interconnection (ISO/OSI). If those words sound big, think of them in terms of what they really mean: "Computer Networking." The first layer is the physical layer. It is concerned with physical parameters of communicating such as frequency, modulation type, and the method used to represent the individual bits that are to be sent. The second layer is the link layer. It is concerned with how data is to be formatted when being transmitted. The third layer is the network layer. It is concerned with getting data from the sending computer (or terminal) to the receiving computer, even if the two stations cannot communicate directly. It must decide how to route data through the network.

How is this different from what you're using now? You can already talk to stations that can't hear you directly using a digipeater. The AX.25 protocol provides for limited level three support. In particular, one can list digipeaters through which a packet is to be relayed. This is called source routing because the routing a packet will take is determined at the source. One problem with this approach is that you have to know the path to the station with which you wish to connect. That's where some of the newer level 3 protocols come in.

You have probably played around with NET/ROM recently. NET/ROM supports its own routing of packets within the NET/ROM system. As a user, you connect to a NET/ROM node, and from there, you can connect to any other NET/ROM node that is reachable. You do not need to know the route that will be taken within the NET/ROM network. NET/ROM uses hop-by-

hop routing. Each NET/ROM node forwards your data to a node that it thinks is one hop closer to the destination. The next node does the same. Nodes do not know the entire route to each destination only the next hop. Further, routes can change as conditions change, or as stations go off the air.

Unfortunately, NET/ROM, and similar level three mechanisms, suffer one problem with respect to routing. As a user, you must know the name of the local node, and the name of the node on the network that is closest to your destination. Knowing the name of your local node is not that much of a problem. Knowing the name of the remote node can be more difficult. To simplify this, mnemonic names are often used in NET/ROM. In other systems, one might name the nodes by area codes or zip codes. Thus, if one knew the address of the final destination, he also knows the name of the nearest node on the network.

With IP, or the Internet Protocol, we require the user to know even less about routing. IP is concerned with getting data from one host to another. Each host has a unique address. The user sends data to the destination address, and it is automatically routed. The user does not need to know the nearest network node to the destination. With NET/ROM and similar mechanisms, the routing is concerned with getting from one point in the communications network to another. With IP, the routing is concerned with end hosts. Within the network, hop-by-hop routing is used. All this does not come for free, however. With IP, there is still the problem of mapping host names to addresses. This can be done using nameservers at well known addresses.

You now know what IP is and how it fits into the network. But, what is TCP/IP? TCP/IP is a suite of protocols. It includes IP at layer three, TCP and UDP at layer four, and many protocols at higher layers. When one thinks of TCP/IP, one thinks of network services, and the ability to easily implement new network services.

TCP, or Transmission Control Protocol, supports the appearance of a reliable connection between two hosts. Applications built on top of TCP include telnet (remote login), SMTP (electronic mail), FTP (file transfer), and many more. Multiple connections may be open to or from any host. Thus, mail or files can be exchanged in the background while the user is connected to the same, or a different system.

Other applications can be built on top of UDP, or Unreliable Datagram Protocol. These datagrams are only unreliable in that they are not guaranteed to get through. Application programs can add timeouts and retransmissions themselves. As such, the user does not see a lack of reliability. Applications built on top of UDP are usually of the type that require a quick response instead of an actual connection. UDP is suited well to distributed applications. One such application is the nameserver used to translate names into IP addresses.

Things become interesting when one starts to consider what new applications can be developed specifically for amateur radio. One application that I think would be useful would be a distributed callbook. When you want to look up a name and address given a callsign, you can query the online callbook. Not all the data will be in one place. There might be a separate registry for each country, or perhaps each call zone. Of course, the user doesn't need to know this. The distributed call book can provide more information than the callbooks we use today. One could allow users to add information about themselves. Perhaps frequencies that they monitor, and when they monitor them.

This article should have given you a better feel for what TCP/IP is, and how it fits in with the existing packet radio network. In future articles I will go into more detail on routing, naming, and applications for the TCP/IP suite of protocols.

PLANNED GROWTH FOR PACKET RADIO IN THE NORTHWEST

by Bob Donnell, KD7NM

The purpose of this document is to put forth my ideas on a planned process for the growth of packet radio networks in the Pacific Northwest.

Implementation would be as required by channel loading.

PHASE I

Phase One of the growth will mainly be involved with improving existing RF paths and providing service to those areas that are currently lacking.

The inter-LAN network will be with current 1200 bits per second modems on simplex frequencies. There will be one or more network back bonechannels. The first backbone exists on 220.95 MHz.

Existing sites that are part of the backbone network are not expected to need any new equipment other than that which will improve their RF reliability.

New systems will need 2 TNCs with NET/ROM installed and the modifications recommended for it, a radio for serving the user LAN, a 220 MHz NBFM radio, and RF support systems for the two radios. These likely include antennas, feed line, cavities, and a circulator or isolater

PHASE II

Phase Two's major improvement is to provide a system for those backbone sites located at natural hubs to tell others that the hub system is currently receiving a valid data carrier so that the other transmitters will hold off rather than transmitting in interference to the station that is currently transmitting.

The proposed method for accomplishing this is to provide a transmitter at the hub site that will transmit a coded signal any time a signal is being heard on the hub systems backbone receiver, and receivers that are set up at the sites that the hub serves to detect that signal and

use it to inhibit the local TNC from transmitting.

The equipment expected to be needed for a hub site is a 450 MHz transmitter with appropriate signal coding equipment and the appropriate supporting RF gear.

The equipment that a spoke site will need is a 450 MHz receiver with the proper decoding equipment to recognize the signal from the hub busy transmitter.

The expected signal to be coded will be an audio tone of such a frequency that it can be rapidly detected so that the inhibit signal will accurately reflect state of the hub receiver. This will be the primary signaling technique. It is anticipated that the hub transmitter will use CTCSS as part of its signal, to be transmitted all the time that the hub transmitter is transmitting, to prevent spoke channel busy receivers from falsely inhibiting their associated transmitters. Use of a CTCSS decoder at the receiver site will be optional based on whether the receiver has intermodulation distortion problems.

PHASE III

Phase Three will be to move the current 1200 bits per second backbone frequency so that it and the proposed 56000 bits per second backbone channel are not in conflict. We have a 100 kHz section of spectrum coordinated for this purpose, from 220.90 MHz to 221.00 MHz.

The 56 kilobits per second system requires only 70 kHz of bandwidth.

I propose changing our current 1200 bits per second backbone frequency from 220.95 MHz to 220.99 MHz. I then propose that the center frequency for the high speed backbone be 220.94 MHz.

This provides a guard band of 10 kHz on either end of our allocated spectrum and a guard band between our high speed backbone and the 1200 bits per second backbone.

The equipment needed to accomplish this will be crystals for those fixed frequency, non-synthesized radios.

PHASE IV

In Phase Four we begin moving to a higher speed backbone by installing 56 kilobits per second RF modems and converters on each high speed site, but using our existing NET/ROM TNCs running at 9600 bits per second.

This is expected to be done on the busiest paths first and will require 3 ports at those sites to support the existing 1200 bits per second back bone until all the other sites in range are using the high speed backbone.

It may also be desirable to keep the 1200 bits per second equipment in place after the change is accomplished, as a backup network, or the equipment moved to a user channel in the same band.

With an eye toward Phase Six as well as the technical advantages of radio systems with rapid T/R switching, using separate receive and transmit converters and a PIN diode antenna switcher should be investigated and seems to be recommended.

The equipment expected to be needed at each site as follows:

RF Modem - input and output at 30 MHz
Receiving converter - 220.94 to 30 MHz
Transmitting converter - 30 MHz to 220.94 MHz
PIN diode antenna switcher
Antenna, feed line, cavities, duplexer, isolater or circulator, as required

PHASE V

The next major upgrade is to change to a controller that can make full use of the 56000 bits per second modems. The current best opinion is that the PS-186 board will become that controller. It has four full-duplex data ports and is expected to be capable of switching packets at data rates up to 1 megabit per second, full duplex, between all ports.

With the changes proposed in Phase Six of this document, it seems to me that any user ports that are operating at 1200 bits per second should be left using the current TNCs running new software. My ideas is to convert the KISS TNC code to support multiple RS-232 addresses in a manner similar to NET/ROM, and allow the networking software in the high speed controller to talk with these TNCs. This approach is only suggested to allow one high speed port on the PS-186 to serve several 1200 bits per second to 9600 bits per second user LANs while leaving as many high speed ports available as possible.

Any ports on the multi-port controller not used for backbone channels can be used to provide user high speed data ports.

The equipment that would be called for in this phase would be the multi port controller, which can be hooked to the already present high speed modems, any new user high speed modems, and to modified KISS equipped TNCs or to the modems in those TNCs.

Other possibly needed equipment is the user port RF gear.

PHASE VI

This step in enhancing the packet network will involve an intense frequency coordination effort. It requires acquiring many wide band frequency allocations on several different bands so that full duplex packet switching can be set up.

This is accomplished by providing each site with a clear frequency to transmit on. The biggest problem with this approach is that it requires a receiving modem and converter for each frequency that is being supported and another multi- port controller for each several converters, at every site.

It will be desirable to provide sufficient frequency isolation between a site and it's neighbors that

while it is transmitting it will be able to listen toall of its neighbors without requiring extensive isolation between the receivers and the transmitter. On bands that are empty enough this will not be a great problem but it may require starting at the 902 MHz or 1230 MHz band.

An alternative for this is to design a hub and spokes network where sites that are not located at natural hubs would only need to receive the hub transmitter frequency but the hub would still be required to support all the spoke transmitter frequencies as well as those for adjacent hubs. In the case of a spoke transmitter, all that would have to be changed to upgrade from Phase Five would be to change the operating frequencies of the port which would likely involve an appropriate converter.

Using the hub and spokes approach also makes the RF problems a little simpler in that the spoke sites would likely all transmit on one band or in one part of a band such that frequencies could be reused, and hub sites would be on other bands or segments for best utilization and least interaction between the hub transmitter and it's receivers.

An example of this idea would place hub A's transmitter on 220 MHz, hub's transmitter on 440 MHz and hub C's transmitter on 902 MHz, approximately. All the spoke stations could be on various frequencies on 1215 MHz, designed for maximum frequency re-use and would listen on the hub transmitter frequency for the hub that serves it.

PHASE VII and Beyond

The crystal ball into the future is cloudy, but setting up 1 megabit per second circuits appears to be out there. There are not currently any modems for this high data rate that I am aware of, but they will come. With high data rates of 56000 bits per second and above available for user access such things as packet digital voice become possible.

Traffic Handling with Packet Radio
by Phil Dunn, KD7ME

Packet radio, with its error checking transmission and store and forward bulletin boards is well suited for handling formal written traffic. The formal message in ARRL format becomes the text or a packet message. Use the command ST NTxx to send traffic, xx being the two letter postal abbreviation for the state or province. When a BBS asks for the subject, enter the city of the addressee followed by the amateur call if a ham. Thus, when the liet command "LT" is used the state, city, and call will be shown. A sample message follows:

NR1 R KD7ME 25 Tacoma WA Nov 19
Joe Ham N7XYZ
123 Main St.
Prescott, AZ 86301
601-345-6789

When a ham accepts a message he relays or delivers the message or if it cannot be done he advises the originating station with reason.
73 BT Phill AR

RS232 - who needs it!!

My packet equipment is working just fine, and what do I need to know about RS232?

If you are running a PK-64, you can safely ignore RS232, and no problems will result. However, if your TNC is one of those which can operate with "any computer having an RS232 port", you may want to know a little more in order to utilize your capabilities. So here is an attempt to shed a little light on the subject.

First, RS232 is actually just a definition of the signals used in serial communication. What is THAT, and so what? Well, serial communication is a means of sending and receiving information, one bit at a time. If we are doing it by electrical means, it requires only two wires — a signal wire, and a return. It is a little like the system deep sea divers have used - one quick tug on the signal rope has a special meaning, two quick tugs another meaning, etc.

This is all very well, but how does it affect packeteers. Well, we are using two serial communmication schemes simultaneously when we are using packet. The first, and most visible, is the packet system itself. Packet is a very sophisticated serial communication technique. The second is the means by which the TNC sends and receives information to and from the computer terminal - the one you are typing on. That second link is the one that uses the RS232 protocol we are concerned with here.

Now, if you are still with me, let's look at that link more carefully. First, could we use a single wire (and return) to connect the TNC to the terminal (computer)? Sure, but it would be a little unhandy. In order to operate our packet, we must be able to both send and receive information between TNC and our terminal (here after referred to as the computer). And in fact, we would like to be able to send and receive at the same time. For example, if we need to tell the TNC to stop sending information, we may not be able to wait until the path is clear, that is, the TNC has stopped sending.

The RS232 protocol allows us to resolve this problem very easily by the simple method of providing TWO serial channels, one for each direction (send, and receive). Now, we can talk to the TNC from the computer at the same time the TNC is talking to the com-

puter. Simple enough, and it has cost us only one additional wire.

We now have a three wire connection (SEND, RECEIVE, and return) between TNC and computer. This is the MINIMUM wiring from TNC to computer. If your equipment uses the industry standard DB25 connectors (25 pins), the pin numbers are:
Pin 2 - TXD - Transmit Data Pin 3 - RXD - Receive Data
Pin 7 - Signal Ground
Besides establishing a standard set of wire functions (TXD, etc.), RS232 specifies the voltage levels for the two logic levels. The two logic levels are "0" and "1", or FALSE and TRUE. By RS232 definitions, a voltage level between -3 and -25 volts is "0", and a voltage level between +3 and +25 volts is "1".

Now we are ready to address the next matter with which we should concern ourselves, namely, FLOW CONTROL. When sending data from the computer to the TNC for transmission to the receiving station, how does the system deal with the problem of making sure the TNC is not asked to take more information than it can handle? The computer is capable of sending data to the TNC at a much higher rate than the TNC can send the data out over the air. The TNC has some memory space, called a buffer, into which the data can be placed while it is waiting for transmission on the radio. However, this buffer can only accomodate a limited amount of data, and would soon be filled if the computer sends much information.

If we are using the simplest possible TNC to computer connection, then the FLOW CONTROL is handled with software. When the TNC buffer is nearly full, it sends a character to the computer which tells the computer to suspend the data flow to the TNC. This character is called XOFF, and is usually a control-S (ascii 19, decimal). Upon receipt of this character, the computer goes into a data hold mode until it receives further instructions. When the TNC buffer is nearly empty, it sends another character to the computer, called XON. This character tells the computer to resume transmission of data to the TNC, and is usually a control-Q (ascii 17, decimal).

This system is actually a two-way street, so that the computer can also tell the TNC that is needs a little time to handle the data flowing from TNC to computer. When both the computer and the TNC understand and use the XON-XOFF characters, data flow can be easily controlled.

So what else is there to know about RS232 and you? Well, the software flow control system just described works fine if your only concern is sending and receiving ordinary text files - containing only standard letters and symbols. And of course, the terminal software you use at your computer must be able to accommodate the XON-XOFF system (most terminal programs do but you may have to tell the program you want to use the XON-XOFF system when you install it). But what if you want to handle files which contain a lot of non-text data? Now, a problem can arise because the data may contain the XON-XOFF characters, and great confusion can result. We must then implement a different FLOW CONTROL scheme, and there is another available. If we add two more wires between the TNC and computer, we can use these two wires to control data flow. One of these wires would tell the computer when it is OK to send date to the TNC, and the other would tell the TNC when it is OK to send data to computer. With this scheme, we now have HARDWARE FLOW CONTROL, which of course can handle any and all characters without any trouble. The two additional wires are called RTS, request to send, and CTS, clear to send. RTS is the signal from the computer to the TNC which tells the TNC that the computer is ready to receive data, and CTS is the signal from the TNC which tells the computer that the TNC is ready to receive data.
The two pins on the DB25 connector are:
Pin 4 - RTS - Request to send
Pin 5 - CTS - Clear to send
By connecting these pins together from the TNC to the computer, we have HARDWARE FLOW CONTROL, and we do not need to use the XON-XOFF SOFTWARE FLOW CONTROL

If we elect to use this method, we must tell the TNC that we have done so with an appropriate command, and we MAY have to inform the computer terminal program that we are using HARDWARE FLOW CONTROL, instead of XON-XOFF FLOW CONTROL. With my PK-232 controller, the command is XFLOW OFF, but consult your TNC manual.

While this does not pretend to tell you all about RS232, I hope that it will be useful to packeteers who want to be sure they are using their capabilities to the fullest extent possible.
73 - Cal Morgan, W7SFT

TNC2 mods for 19.2kb operations

Here is a fairly easy way to modify a TNC2 or clone for 19.2k baud operation over the rs-232 serial port. This modification is primarily of concern to people running higher modem rates with an external modem (bypassing the internal 1200 baud modem) like the AEA 19.2kb radio modem or the WA4DSY 56kb rf modem.

Another interesting application is to improve the response time of a bbs or a TCP/IP station.

These modifications apply to TNC-2s or exact clones. In general, these modifications are accurate for the MFJ 1270 and the PAC-COMM TNC-200 although the parts designations may not be accurate. These modifications may not work on non-clones such as the PaComm TNC-220, the MFJ-1274, the current AEA units

or others that have deviated from the TNC-2 design. Of course with careful thought, any of the excluded units can be made to work.

This text was lifted directly from a text written by WA4VZW and WD4OQC of the WA4DSY 56kb radio modem project.

Preliminary steps:

Open the case to the TNC and remove the printed circuit board (PCB). The TNC-2 requires that the front panel and the philips head screw on the voltage regulator be removed. For the MFJ unit, remove the 4 case screws and the 2 front panel screws. Then remove the 4 screws that hold the PCB to the bottom case half.

1. Cut trace between the terminal baud rate switch pin 1 and IC U1, pin 2. Install a jumper between the switch pin 1 and IC U1, pin 10

2. Replace U3, LM324n with a TL084 chip.

Wrap-up

Inspect your work for solder bridges or cold joints. Remove any soldering residue using alcohol or acetone and a brush. Thoroughly dry the board. Replace the PCB in the case. For the TAPR-style TNC's, coat the back side of the LM7805 regulator with heat sink grease before reattaching it to the case. On MFJ units, make sure the clip-on heat sink makes firm contact and insure that there is adequate heat sink grease on the clip-to-chip AND clip-to-grease joints.

If after doing this modification, and you find that the unit doesn't run at 19.2kb, make sure that the computer connected to the TNC is running at 19.2kb. It is also possible that at least the Z80 SIO and possibly the Z80 itself will have to be upgraded from the A parts that are currently in the unit to B grade parts. Another possibility is that the TL084 installed is not up to spec. Trying another part might solve the problem.

Disclaimer: The author hasn't actually tried this modification yet, but will be doing this with some other modfications to make the TNC work with 56kb radio modems. More about this and other related items will appear in future issues of Zero Retries.

73 de N7AKR Bob

NAPRA BOARD MEMBERS

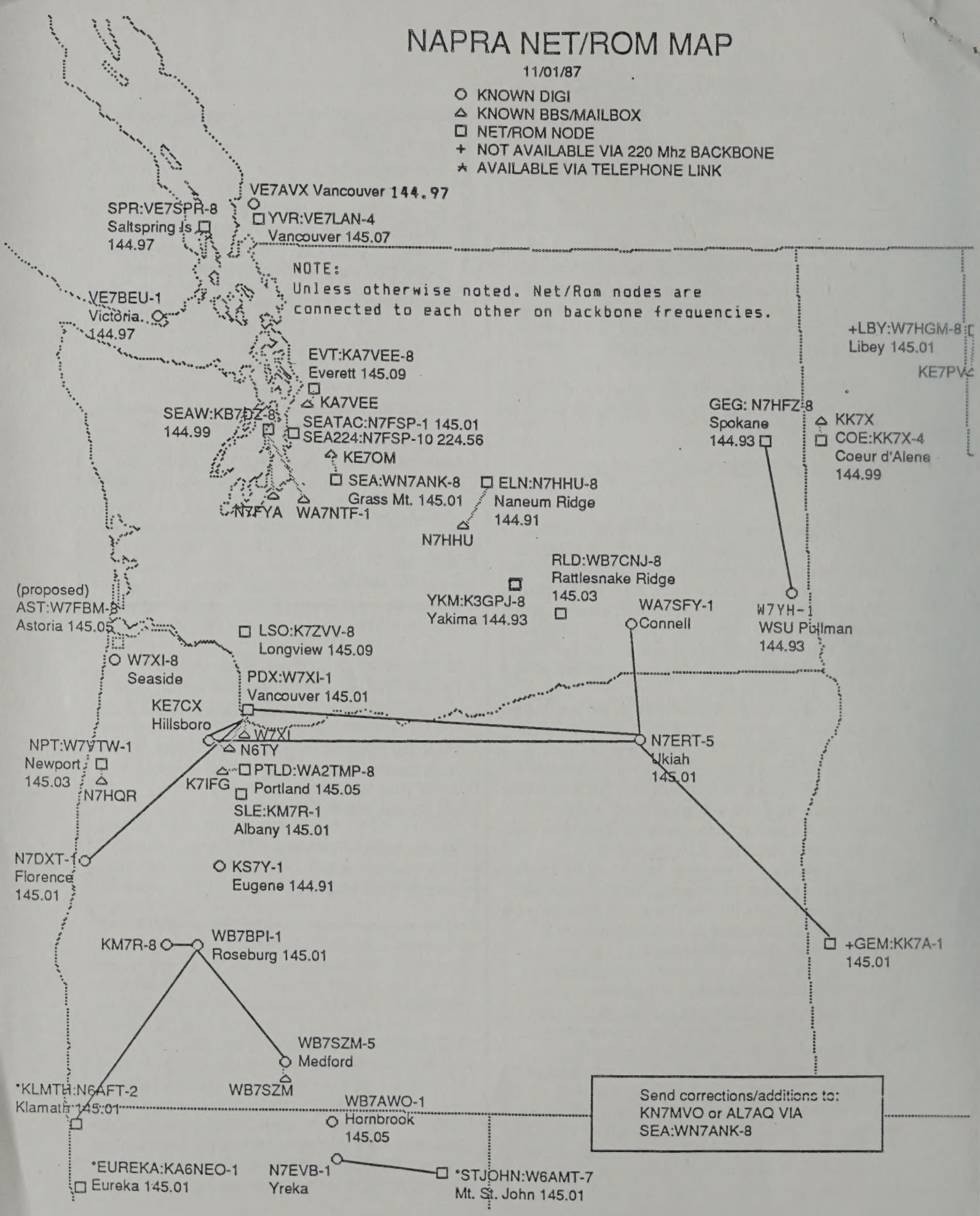
Pres.: Budd Churchward, WB7FHC (206)857-5271
Exec VP: Don Werts, KA6QMT (206)228-1099
Tech. VP: Bob Donnell, KD7NM (206)827-8005
Sec.: Ren Roderick, WA7QFR (206)488-3171
Treas.: Peter Dahl, WA7FUS (206)839-7717
Board: Larry Babcock, WB7USL (206)246-8715
Board: Cliff Neuman, N1DMM (206)789-8788

NAPRA Dues

Annual dues are $15.00 which may be paid to the treasurer at NAPRA meetings or by mail. The mailing address for dues is:
NAPRA c/o
Peter Dahl
1609 South 261st Place
Kent, Wa. 98032

Other correspondence addresses for NAPRA:
Budd Churchward
13304 131st St. KPN
Gig Harbor, Wa. 98335

Ren Roderick
14824 140th Ave. NE
Bothell, Wa. 98011


NAPRA NET/ROM MAP
11/01/87
KNOWN DIGI
KNOWN BBS/MAILBOX
NET/ROM NODE
+ NOT AVAILABLE VIA 220 Mhz BACKBONE
* AVAILABLE VIA TELEPHONE LINK
VE7AVX Vancouver 144.97
SPR:VE7SPR-8
Saltspring Is
144.97
YVR:VE7LAN-4
Vancouver 145.07
NOTE:
Unless otherwise noted. Net/Rom nodes are
connected to each other on backbone frequencies.
VE7BEU-1
Victoria
144.97
+LBY:W7HGM-8
Libey 145.01
KE7PV
EVT:KA7VEE-8
Everett 145.09
KA7VEE
SEAW:KB7DZ-8
144.99
SEATAC:N7FSP-1 145.01
SEA224:N7FSP-10 224.56
KE7OM
SEA:WN7ANK-8
Grass Mt. 145.01
N7FYA
WA7NTF-1
ELN:N7HHU-8
Naneum Ridge
144.91
N7HHU
GEG: N7HFZ-8
Spokane
144.93
KK7X
COE:KK7X-4
Coeur d'Alene
144.99
RLD:WB7CNJ-8
Rattlesnake Ridge
145.03
YKM:K3GPJ-8
Yakima 144.93
WA7SFY-1
Connell
W7YH-1
WSU Pullman
144.93
(proposed)
AST:W7FBM-8
Astoria 145.05
LSO:K7ZVV-8
Longview 145.09
W7XI-8
Seaside
PDX:W7XI-1
Vancouver 145.01
KE7CX
Hillsboro
W7XI
N6TY
NPT:W7YTW-1
Newport
145.03
N7HQR
K7IFG
PTLD:WA2TMP-8
Portland 145.05
SLE:KM7R-1
Albany 145.01
N7ERT-5
Ukiah
145.01
N7DXT-1
Florence
145.01
KS7Y-1
Eugene 144.91
KM7R-8
WB7BPI-1
Roseburg 145.01
+GEM:KK7A-1
145.01
WB7SZM-5
Medford
*KLMTH:N6AFT-2
Klamath 145.01
WB7SZM
WB7AWO-1
Hornbrook
145.05
Send corrections/additions to:
KN7MVO or AL7AQ VIA
SEA:WN7ANK-8
*EUREKA:KA6NEO-1
Eureka 145.01
N7EVB-1
Yreka
*STJOHN:W6AMT-7
Mt. St. John 145.01

NAPRA
1609 South 261st Place
Kent, Wa. 98032

ZERO RETRIES